Amendments To The Specification:

On page 4, please delete the paragraph [0007], and replace it with the following paragraph:

[0007] Turning back to Fig. 1, The storage systems 20, 21 also are connected with a high speed communication link 30, which will be described. The data stored on data storage system 20 may be crucial to the operation of the computer 12. Therefore, a contingency solution is desirable in the event that a problem is experienced with storage system 20 to ensure that the data is not lost and to minimize the risk of computer 12 being down or unavailable due to issues with the storage system 20. Potential problems with the storage system 20 can include hardware and/or software errors that may make stored data unrecoverable, as well as catastrophic events such as an earthquake, power shortage or other natural disaster that could result in the destruction or unavailability of the storage system. One solution for protecting the data stored in storage system 12 is to mirror the data into another storage system, such as storage system 13, as will be described.

On page 10, please delete the paragraph [0024], and replace it with the following paragraph:

[0024] The Protocol Field 78 indicates that the next level protocol used in the data portion of the internet diagram. The Header Checksum 80 is a checksum on the header only. For those header fields that change (i.e. Time to Live) this is recomputed and verified at each point the internet header is processed. The Source IP Address 82 indicates the source address of the sending computer, while the Destination IP Address 84 indicates the destination or

target address of the receiving computer. The Options Field 86 is for the options that may or may not appear in datagrams. Generally, they are implemented by all IP modules, including host computers, but in any particular datagram their transmission optional. Some of the options include Security, Record Route, Stream Identifier, Internet Timestamp, End of Option List, No Operation, Loose Source and Record Route, Strict Source and Record Route, Record Route and Stream Identifier. Lastly, the Padding field 88 is used to ensure that the internet header ends on the 32 bit boundary. If the datagram does not end on a 32 bit boundary, padding or extra bits may be added to the datagram to ensure that the header ends on the desired boundary.

On page 11, please delete the paragraph [0026], and replace it with the following paragraph:

[0026] Turning back to Fig. 1, data storage systems 20, 21 will now be described in further detail. In this example primary data storage system 20, in this example, includes front end adapters 16a-n which interfaces with the computer 12 and receives data from the computer 12 over a communication bus 18. Data received from the computer 12 is typically stored in cache 24 before being transferred through storage bus adapters or disk adapters 26a-n over an internal bus 31 to the primary storage devices 28a-n. The storage devices may include physical devices such as disk drives, optical disks or other storage devices. The plurality of data storage devices on all of the storage systems in the system, may have various volumes (both logical and physical).

On page 12, please delete the paragraph [0028], and replace it with the following paragraph:

[0028] Primary data storage system 20, according to one embodiment of the invention, also includes a service processor (not shown) coupled to primary data storage system 20, which can provide monitoring, repair, service or status access to the storages system. The storage system 20 also may include a secondary disk adapter 48. Secondary disk adapter 48 is coupled via the high speed communication link 30 to the other secondary disk adapter 49, possibly remote data storage system 21.

On page 13, please delete the paragraph [0030], and replace it with the following paragraph:

[0030] As in the case of the primary data storage system 20, the secondary data storage system 21 includes, in addition to the secondary data storage system controllers containing host bus adapters 23a-n and storage bus adapters 27a-n, a storage devices, 29a-n. As with the primary data storage system 20, the storage devices 29a-n may have various volumes and usages, in particular where 28a for example, is a primary storage device for storage system 20 and 29a which is a primary storage device for storage system 21. As in the case of the primary data storage system 20, the secondary data storage system 21 also may include a secondary disk adapter 49.

On page 13, please delete the paragraph [0031], and replace it with the following paragraph:

[0031] It should be understood that each physical storage device in storage system may also include a secondary volume (not shown) which serves as secondary storage for the primary data stored on volume of the primary data storage device 28a-n. Similarly, the primary data storage system 20 may include a secondary storage volume which stores primary data received from and copied from the secondary storage system 21. For purposes of this description, primary storage volumes are referred to as (R1) volumes, while secondary volumes are referred to as (R2) volumes (not shown) in Fig. 3 shown as 40. Additionally, each storage device in the storage system may include one or more local volumes, which are, in normal operating fashion, accessed only by their locally connected computer.

On page 14, please delete the paragraph [0032], and replace it with the following paragraph:

[0032] The secondary storage system 21 also includes at least one a front end adapter 23 which may receive data from a connected secondary computer 13. Optionally, the primary host computer 12 may include a data signal path (not shown) to connect directly into a front end adapter 23 of the secondary data storage system 21. Additionally, the secondary host computer 13 may include a data signal path (not shown) directly into a host bus adapter 22 of the primary data storage system 20.

On page 16, please delete the paragraph [0039], and replace it with the following paragraph:

[0039] Before proceeding further, in addition to the remote mirroring previously described, the present invention can also be used with any process or system that allows for the duplication or mirroring of a set of data or volumes (both logical and physical). U.S. Patent Number 6,101,497 entitled Method and Apparatus for Independent and Simultaneous Access to a Common Data Set describes a way to have concurrent access to a data set at a single site within a single storage system. In other words, instead of remotely mirroring the data as previously described, Business Continuance Volumes (BCVs) are created. This enables a first computer connected to a storage system to work with or respond to one type of application, such as the aforementioned OLTP, while another computer also connected to the storage system could work with or respond, using the same data, to another application such as a backup or other OLTP application. Fig. 7 shows a representation of a single storage system 20 shown being connected to a plurality of computers 12 and 13. As described in U.S. Patent No. 6,101,497 a series of steps or commands are used to set up the BCV volumes 41 a-n, use then while making certain data integrity is maintained. For purposes of the present application, it is sufficient to understand that the data is copied from one volume, such as 26a, to a BCV volume 41a, so the data in the BCV volume can be used by a second computer or application.

On page 19, please delete the paragraph [0043], and replace it with the following paragraph:

[0043] The software that runs on computers 12, 13 to accomplish the operations of the present invention, is comprised of a single processes. In the preferred embodiment of the

invention, the process is a kernel-mode driver, which needs to simulate the functionality of a network card, so that as far as the applications are concerned there is another NIC in the system. A network card, often called a network interface card (NIC) is usually an expansion board that is included with a computer so that the computer can be connected to a network. Most NIC's are designed for a particular type of network, protocol and media, although some can serve multiple networks. The process when installed onto the computer must be, transparent to the computer and the other components in the system 10. In other words, the software must be indistinguishable from a real NIC. For illustrative purposes only, this process will be referred to as a Virtual Network Interface Card (VNIC) process. Fig. 5 is a logical representation demonstrating the operating environment of the VNIC process within a Windows NT operating system shown as 90. In one embodiment of the invention the VNIC process is implemented as a Network Driver Interface Specification (NDIS) miniport driver. Typically, the primary purpose of NDIS is to define standard Application Programming Interfaces (APIs) for NICs. Since a NIC is normally a piece of hardware, the implementation of such is wrapped with some sort of device driver, such as a media access controller (MAC) so that all NICs for the same media (i.e. Ethernet) can be accessed using a common programming interface. NDIS also provides a library of functions, sometimes called a wrapper, that can be used by device drivers as well as higher level protocol drivers (such as TCP/IP). A miniport driver typically replaces platform-specific functionality with platform independent "wrapper" functions. Therefore, in one embodiment of the invention, the VNIC process is implemented as an NDIS miniport driver that simulates the existence of an Ethernet adapter. Since the VNIC process is a kernel based piece of the operating system it will be

called on many different threads (or programs) executing simultaneously. Because the VNIC process must respond to these externally generated events, it spins up an internal thread, to provide synchronized access to critical data performing time sensitive tasks such as Input/Output (I/O) operations.

On page 20, please delete the paragraph [0044], and replace it with the following paragraph:

[0044] Fig. 5 shows the VNIC process 116 within the datalink layer in a Windows NT operating environment, which is shown as 100. By datalink it is meant the layer that governs access onto the network and reliable transfer of packets across the network. As can be seen the VNIC process operates entirely within the kernel of the operating system. As such it interacts with the computer 12 or 13 of Fig. 1 entirely through NDIS-defined functions and protocols. The NDIS specification which is fully incorporated herein by reference is entitled Network Driver Interface Specification, version 2.01, published in 1996 by 3COM Corporation and Microsoft Corporation. Above the datalink layer 100 is the Transport Driver Interface 120 which serves as the interface between higher lever drivers such as those shown as clusdisk.sys 92, that in a clustered application is responsible for ensuring only one node at a time can access a cluster disk, and diskclass.sys 94 that presents logical representations of the disk drives to upper layers in the system via an internal communication mode as shown as 98. In this example, the communication mode is shown as a Transmission Control Protocol over IP suite (TCP/IP). In this example, data comes through the datalink layer through the TDI using TCP/IP, and then can be passed into the NDIS 108 (which includes the other components,

such as LAN 102, Native Media Aware Protocol 104, and NDIS Intermediate 106), where depending on destination (i.e. network path) of the data can either head to the network through NDIS miniport 110 and NIC 112 or to the storage system 20 through region shown as 114, which includes VNIC 116 and NIC 118. However, once the data arrives in the NIC 118 or 112, it is placed into the appropriate NIC's buffer for transmission onto the physical medium.